

**Amendments to the Claims:** This listing of claims will replace all prior versions, and listings, of claims in the application

Listing of Claims:

1. (Currently Amended) A control device for electrical or electronic equipment, the device having processing means and non-volatile memory means, the non-volatile memory means having installed programs executable by the processing means ~~direct~~ directly from the non-volatile memory means, each program being made up of processing elements at least one of which can be modified or upgraded by the installation of a patch, ~~characterised in that~~ wherein:  
a part of the memory means is used as a patch registry containing a list of patch descriptor elements, and  
the processing means is arranged to install a new patch by modifying the program processing element to which it relates and storing a descriptor element for the patch in the patch registry.
2. (Currently Amended) A device as claimed in claim 1 in which the patch registry includes information relating to ~~the progress of the installation of a~~ the new patch.
3. (Currently Amended) A device as claimed in claim 1 or 2 in which the patch registry includes a list of unused program memory blocks for each ~~processor~~ processing element.
4. (Original) A device as claimed in claim 3 in which, on installation of a new patch, unused program memory in the list is used to extend the patch registry to contain information relating to the new patch.
5. (Currently Amended) A device as claimed in ~~any preceding claim~~ 1 or 2 in which each patch descriptor element contains a text description of the patch ~~which can~~ configured to be presented to a user interface.
6. (Currently Amended) A device as claimed in ~~any preceding claim~~ 1 or 2 in which each patch descriptor element contains a list of modified code descriptor elements.

7. (Currently Amended) A device as claimed in claim 6 in which the modified code descriptor elements identify the ~~processor~~processing element to which the patch has been applied.

8. (Currently Amended) A device as claimed in claim 7 in which the modified code descriptor elements identify the a start address of a faulty code block in the ~~processor~~processing element.

9. (Currently Amended) A device as claimed in claim 7 ~~or 8~~ in which the modified code descriptor elements identify the a number of bytes of faulty code in the ~~processor~~processing element being repaired by the patch.

10. (Currently Amended) A device as claimed in claim 7, ~~8 or 9~~ in which the modified code descriptor elements include the a start address of the memory area used for the repaired code contained in the patch.

11. (Currently Amended) A device as claimed in ~~any claims 7 to claim~~ claim 10 in which the modified code descriptor elements contain information in the form of binary flags describing how the repaired code contained in the patch was installed.

12. (Currently Amended) A method of modifying programs installed in a control device for electrical or electronic equipment, the control device having processing means and non-volatile memory means, the non-volatile memory means having installed programs executable by the processing means ~~direct~~directly from the non-volatile memory means and each program being made up of processing elements, the method comprising:

- a) downloading to the control device a patch from an external source containing code for modifying a one of the program processing elements,
- b) installing the patch by modifying the one program processing element to which it relates in the non-volatile memory; and
- c) storing a descriptor element for the patch in a separate part of the non-volatile memory designated as patch registry.

13. (Currently Amended) A method as claimed in claim 12 including, during step b), the step of storing, in the patch registry, information relating to the progress of the installation of a new patch.

14. (Currently Amended) A method as claimed in claim 12 or 13 additionally comprising the step of storing in the patch registry a list of unused memory blocks for each ~~processor~~ element of the processing elements.

15. (Original) A method as claimed in claim 14 in which, on installation of a new patch, the patch registry is extended using unused memory and information relating to the new patch is stored in said unused memory added to the patch registry.

16. (Currently Amended) A method as claimed in ~~any of claims claim 12 to 15~~ or 13 further including the step of configuring each patch descriptor element to contain a text description of the patch which ~~can~~ is configured to be presented to a user interface.

17. (Currently Amended) A method as claimed in ~~any of claims claim 12 to 16~~ or 13 further including the step of configuring each patch descriptor element to contain a list of modified code descriptor elements.

18. (Currently Amended) A method as claimed in claim 17 in which the modified code descriptor elements are configured to identify the ~~processor~~ one processing element to which the patch has been applied.

19. (Currently Amended) A method as claimed in claim 17 in which the modified code descriptor elements are configured so as to identify ~~the~~ a start address of a code block in the ~~processor~~ one processing element to be modified.

20. (Currently Amended) A method as claimed in claim 18 ~~or 19~~ in which the modified code descriptor elements identify ~~the number~~ respective numbers of bytes of code in the ~~processor~~ one processing element being modified by the patch.

21. (Currently Amended) A method as claimed in claim 18, ~~19 or 20~~ in which the modified code descriptor elements include ~~the~~ a start address of ~~the~~ a memory area used for the modified code contained in the patch.
22. (Currently Amended) A method as claimed in ~~any of claims~~ claim 18 ~~to 21~~ in which the modified code descriptor elements contain information in the form of binary flags describing how ~~the~~ repaired code contained in the patch was installed.
23. (Currently Amended) A method as claimed in ~~any of claims~~ claim 12 ~~to 21~~ in which step (b) comprises overwriting code in ~~a~~ the one processing element with code contained in the patch.
24. (Currently Amended) A method as claimed in ~~any of claims~~ claim 12 ~~to 21~~ in which step (b) comprises installing the patch code in a selected unused part of the non-volatile memory space and diverting program flow to ~~this~~ the selected part of the non-volatile memory space and back again thereby bypassing code in the unmodified processing element.